

Power Allocation for Cognitive Wireless Mesh Networks by Applying Multi-agent Q -learning Approach

Xianfu Chen^{†‡}, Zhifeng Zhao^{†‡}, and Honggang Zhang^{†‡}

[†]York-Zhejiang Lab for Cognitive Radio and Green Communications

[‡]Department of Information Science and Electronic Engineering(ISEE)

Zhejiang University, Zheda Road 38, Hangzhou 310027, China

Email: {chenxianfu, zhaozf, honggangzhang}@zju.edu.cn

Abstract

As the scarce spectrum resource is becoming over-crowded, cognitive radios (CRs) indicate great flexibility to improve the spectrum efficiency by opportunistically accessing the authorized frequency bands. One of the critical challenges for operating such radios in a network is how to efficiently allocate transmission powers and frequency resource among the secondary users (SUs) while satisfying the quality-of-service (QoS) constraints of the primary users (PUs). In this paper, we focus on the non-cooperative power allocation problem in cognitive wireless mesh networks (*CogMesh*) formed by a number of clusters with the consideration of energy efficiency. Due to the SUs' selfish and spontaneous properties, the problem is modeled as a stochastic learning process. We first extend the single-agent Q -learning to a multi-user context, and then propose a conjecture based multi-agent Q -learning algorithm to achieve the optimal transmission strategies with only private and incomplete information. An intelligent SU performs Q -function updates based on the conjecture over the other SUs' stochastic behaviors. This learning algorithm provably converges given certain restrictions that arise during learning procedure. Simulation experiments are used to verify the performance of our algorithm and demonstrate its effectiveness of improving the energy efficiency.

Index Terms

cognitive radio, cognitive wireless mesh networks, dynamic spectrum access, power allocation,

I. INTRODUCTION

In wireless communications, the electromagnetic radio frequency is the most precious resource, the use of which is regulated by governmental agencies on a long-term basis for large geographical regions. Currently, the frequency band is overcrowded and there hardly exists space available for the emerging wireless services. However, on the other hand, we are increasingly beginning to see that the fixed spectrum allocation policy has resulted in vastly under-utilized spectrum holes. In November 2002, the Federal Communications Commission (FCC) published a report which shows up to 70% of the allocated spectrum in certain measurement geographical areas in the United States are idle in most of the time [17]. The limited available spectrum and the inefficiency in the spectrum usage necessitate a new communication paradigm to exploit the existing wireless spectrum opportunistically [1]. New approaches such as opportunistic spectrum access (OSA) and dynamic spectrum access (DSA) are proposed to bridge the enormous gulf in time and space between the regulation and the potential spectrum efficiency. CR is a promising radio technique possessing intrinsic capability to exploit these spectrum holes by sensing a wide range of the frequency bands and identifying currently unused spectrum blocks, and then communicating by an opportunistically overlaying manner [9], [16].

Up to now, the research on CR has already penetrated into different types of wireless networks, and covered almost every aspect of wireless communications [2], [12], [19], [22], [25]. In this paper, we focus our emphasis on the cognitive wireless mesh networking scenario, named as *CogMesh* as described in our previous work [2]. One of the critical challenges in deploying *CogMesh* is how to design an efficient power allocation scheme for the usage of detected available 'spectrum holes' among the SUs while achieving interference-tolerable spectrum sharing with the neighboring PUs. An efficient design is to maximize the network performance subject to guaranteeing the PU transmissions and the signal-to-interference-plus-noise ratio (SINR) of the SUs' ongoing connections. The transmission power is a 'double-blade' sword. On one hand, the higher the transmission power, the better performance a SU can expect; on the other hand, this better performance is obtained at the expense of not only causing higher interference to both the PUs and the other SUs, but also increasing power consumption. In wireless networks, the choice of transmission power fundamentally affects the performance of multiple protocol

layers. Recently, there has been much work on formulating the power allocation problem with cross layer design. The interested reader is referred to [10] and cited references therein. But they assume that the users are cooperative. Thus, the cross layer design problem can be converted to the systems optimal design. In our considered *CogMesh* scenario, cooperation among the neighboring clusters helps to quantify the tradeoff, for example if a central entity controls the signaling in the network, it can update and broadcast the relevant information to all clusters and their registered SUs.

However, it's more suitable to address the power allocation of *CogMesh* within a non-cooperative game-theoretic framework, since there are conflicting interests among the clusters. [6] considered non-cooperative energy efficient spectrum access for a wireless CR ad hoc network by combining an unconstrained optimization method with a constrained partitioning procedure. [25] studied the distributed multi-channel power allocation for CR networks with strategy space to address both the co-channel interference among SUs and the interference temperature regulation imposed by PUs. In [22], Fan et al. proposed a price based spectrum management scheme for CR networks. Assuming that SUs repeatedly negotiating their best transmission powers and spectrum, SUs announce prices to reflect their sensitivities to the current interference levels, and then adjust their transmission powers. Our work originates from this non-cooperative problem, whereas we propose a reinforcement learning algorithm in this paper to deal with it.

In order to formulate the non-cooperative game theoretically, we first model the self-interest property of power allocation in *CogMesh*. Generally, the concept of reward refers to the level of satisfaction the decision-maker receives as a return of its performed action. We construct a reward function with the consideration of energy-efficiency. Based on the reward function, we model the selfish behaviors as a non-cooperative power allocation game, that is, each SU maximizes its own reward, regardless of what all the other SUs do. In spite of this selfish nature, it is significant for the SUs to adapt to the environment changes since energy efficiency is highly dependent on environmental factors like primary users' behavior patterns and traffic QoS requirement.

Therefore, we formulate the power allocation in *CogMesh* as a stochastic learning process [4], [11], [20], [26] featured by non-cooperative game playing among the local clusters, in which the SUs are spontaneous rational players with advanced learning capability; but the SUs may be selfish at some extent. Then we adopt the framework of reinforcement learning known as Q -learning [20] in this paper. As illustrated in Fig. 1, during the learning procedure, the SU

updates its strategy according to its experience with different actions without explicit modeling of the environment. Based on the single-agent Q -learning algorithm, a multi-agent Q -learning is proposed to accomplish the problem of multi-user stochastic learning. One challenge of the proposed approach to our scenario is that the SUs do not know the information of other SUs due to the non-cooperation among clusters. Then the networking environment is non-stationary for all SUs and the convergence of learning process may not be assured. To alleviate the lack of mutual information exchange, the SUs form internal conjectures over how the other SUs react to their present actions with only local observations from direct interactions with the *CogMesh* environment. Learning is finished asymptotically by appropriately making the use of past experience. Essentially, our argument is that every rational SU has the motivation to improve its performance even if they are selfish by nature.

Some work about reinforcement learning in CR networks have been investigated [3], [14], where the studies are focused on the channel allocation, which is different from the topic in this paper. Our work is the first one toward exploring the multi-agent Q -learning theory in the stochastic non-cooperative power allocation game in CR networks, especially, *CogMesh*. Compared to the previous work, this work provides the following three key insights:

- Firstly, for the non-cooperative power allocation game in *CogMesh*, we show that the selfish dynamics exist in the stochastic learning process.
- Secondly, we present a reinforcement learning algorithm where the update rule is based on SU's own private and incomplete information; the selfish learning dynamics converge.
- Thirdly, this paper also contributes to the general literature on multi-agent Q -learning. While traditional multi-agent Q -learning algorithms, such as Nash- Q [11] and CE- Q [8] in computer science (CS), rely on the full information of all agents in the environment. This is impossible in the scenarios of wireless communication, since there exist conflicting interests among the users. Thereupon, we developed a conjecture-based multi-agent Q -learning.

The rest of this paper is organized as follows. In the next section, we briefly introduce the single-agent Q -learning algorithm and its extension to multi-agent scenarios. In Section III, we formulate the non-cooperative power allocation problem as a stochastic learning game, for which we also present the design objective and the relevant challenging issues. In Section IV, we propose a conjecture-based multi-agent Q -learning algorithm; and the convergence of the

proposed algorithm is further investigated. The numerical results are included in Section V, verifying the validity and efficiency of the proposed algorithm. Finally, we present in Section VI a conclusion of this paper.

II. PRELIMINARIES OF Q -LEARNING ALGORITHM

In this section we first give a brief introduction on the single-agent Q -learning algorithm, and then extend the algorithm to multi-agent scenarios. Our description adopts standard notations and terminologies from the framework of reinforcement learning [7], [11], [20].

A. Single-agent Q -learning

The environment, which an agent interacts with, is typically formulated as a finite-state Markov Decision Process (MDP). Let \mathcal{S} be a discrete set of environment states, and \mathcal{A} a discrete set of actions. At each step t , the agent senses the state $s^t = s \in \mathcal{S}$ and selects an action $a^t = a \in \mathcal{A}$ to perform. As a result, the environment makes a transition to the new state $s^{t+1} = s' \in \mathcal{S}$ according to probability $T_{ss'}(a)$ and thereby generates a feedback (reward) $r^t = r(s^t, a) \in \mathbf{R}$ passing to the agent. This process is repeated infinitely.

The task of the agent is then to learn an optimal policy $\pi^*(s)$ for each s , which maximizes the total expected discounted reward over an infinite steps.

$$V^\pi(s) = E \left[\sum_{t=0}^{\infty} \beta^t r(s^t, \pi(s^t)) | s^0 = s_0 \right], \quad (1)$$

where s_0 is the initial state, E means the expectation operator and $\beta \in [0, 1)$ is the discount factor. We can rewrite Equation (1) as [20]

$$V^\pi(s) = E[r(s, \pi(s))] + \beta \sum_{s' \in \mathcal{S}} T_{ss'}(\pi(s)) V^\pi(s').$$

It has been proven that the optimal policy satisfies the Bellman optimality equation

$$V^*(s) = V^{\pi^*}(s) = \max_{a \in \mathcal{A}} \left\{ E[r(s, a)] + \beta \sum_{s' \in \mathcal{S}} T_{ss'}(a) V^*(s') \right\}. \quad (2)$$

One of the attractiveness of Q -learning is that it assumes no a prior knowledge about the state transition probabilities $T_{ss'}(a)$. We define the right-hand side of Equation (2) by,

$$Q^*(s, a) = Q^{\pi^*}(s, a) = E[r(s, a)] + \beta \sum_{s' \in \mathcal{S}} T_{ss'}(a) V^{\pi^*}(s'). \quad (3)$$

Then by Equation (2),

$$V^*(s) = \max_{a \in \mathcal{A}} Q^*(s, a).$$

The optimal state value function $V^*(s)$ can be hence obtained from $Q^*(s, a)$. And in turn, Equation (3) may be expressed as

$$Q^*(s, a) = E[r(s, a)] + \beta \sum_{s' \in \mathcal{S}} \left\{ T_{ss'}(a) \left[\max_{b \in \mathcal{A}} Q^*(s', b) \right] \right\}.$$

In Q -learning, the agent tries to find $Q^*(s, a)$ in a recursive way with the information $\langle s, a, r^t, s' \rangle$. The updating rule is

$$Q^{t+1}(s, a) = (1 - \alpha_t)Q^t(s, a) + \alpha_t \left[r^t + \beta \max_b Q^t(s', b) \right],$$

where $\alpha_t \in [0, 1)$ is the learning rate. Assuming that each action is executed in each state an infinite number of times and the learning rate α_t is decayed appropriately in a suitable way, the $Q^t(s, a)$ will finally converge to $Q^*(s, a)$ with probability (w.p.) 1 as $t \rightarrow \infty$ [23].

B. Multi-agent Q -learning

Consider an N -agent game, each agent is equipped with a standard Q -learning algorithm and learns without any cooperation with the other agents. The received rewards and state transitions, however, depend on the joint actions of all agents. Let \mathcal{S}_i be a discrete set of environment states and \mathcal{A}_i a discrete set of actions relevant to agent i . At each step t , the agent senses the environment state $s_i^t = s_i \in \mathcal{S}_i$, then independently chooses action $a_i \in \mathcal{A}_i$. Consequently, agent i receives $r_i^t = r_i(s_i^t, a_1, \dots, a_N)$ and the environment transits to a new state $s_i^{t+1} = s'_i \in \mathcal{S}_i$ according to the fixed probabilities $T_{s_i s'_i}(a_1, \dots, a_N)$. Note that r_i^t and $T_{s_i s'_i}$ are defined over the joint actions (a_1, \dots, a_N) .

III. PROBLEM FORMULATION

In this paper, we consider a non-cooperative power allocation system in which each SU behaving as a learning agent adjusts its transmission power level based on some reward received from the self-interested *CogMesh* environment to arrive at the optimal strategy. The key component for describing the selfish interest is the reward function. In this section, we first present a reward model for the power allocation, which takes the energy-efficiency into account. Based on the reward model, we formalize the power allocation problem through the non-cooperative

game playing. Following that, we convert the non-cooperative playing into a stochastic learning process. Finally, we discuss the design objective and highlight the challenging issues.

A. Reward Function and Non-cooperative Power Allocation Game

We consider a generalized *CogMesh* networking example consisting of several specific PU links (i.e., primary transmitter PT and primary receiver PR) and one CR network formed by a set $\mathcal{N} = \{1, \dots, N\}$ of SU links spatially distributed in non-overlapping clusters (see Fig. 2). Due to opportunistic spectrum accessing, they coexist in the same area and share the same frequency band with bandwidth W simultaneously. We assume that each user operates in a half-duplex manner, which means it cannot receive any signal when it's transmitting, and vice versa. The total interference plus noise measured by any SU includes PU-to-SU interference, SU-to-SU interference, and the Additive White Gaussian Noise (AWGN). A SU suggests a CR link consisting of a pair of CR nodes, and we use a SU and a CR link interchangeably.

We designate the transmission power and Signal-to-Interference-plus-Noise Ratio (SINR) for SU i by $p_i(p_i^{\min} \leq p_i \leq p_i^{\max})$ and γ_i , respectively. The other SUs' transmit power vector is denoted by $\mathbf{p}_{-i} = (p_1, \dots, p_{i-1}, p_{i+1}, \dots, p_N)$. Assume that the channel gains evolve slowly with respect to the SINR evolution, the SINR of the SU i in this problem formulation is given by

$$\gamma_i(p_i, \mathbf{p}_{-i}) = \frac{h_{ii}p_i}{\sigma + \phi_i^{PU} + \sum_{j \in \mathcal{N} \setminus i} h_{ji}p_j},$$

where h_{ji} is the channel gain between the transmitter of SU link j and the receiver of SU link i , ϕ_i^{PU} denotes the PU-to-SU interference at the receiver of SU link i , and σ is the AWGN power.

The goal of power allocation within the *CogMesh* framework is to ensure that no SU's SINR falls below its threshold γ_i^* chosen to guarantee adequate QoS, i.e.,

$$\gamma_i \geq \gamma_i^*, \forall i \in \mathcal{N}.$$

Furthermore, the opportunistic spectrum access enables the SUs to transmit with overlapping spectrum and coverage with PUs, as long as that the performance degradation induced on the PUs is tolerable. In this paper, we consider the following power mask constraint as in [22], that is, the transmission power level of SU i over the detected frequency band is constrained by

$$p_i \leq p_{mask}, \forall i \in \mathcal{N}, \quad (4)$$

where p_{mask} is the power mask and is given as a priori. Such a hardware based power mask is easier to manipulate at the design stage from a practical point of view. This is because the number of active SUs that share the same spectrum with the PUs varies in time and space, it is impossible to design the device to account for a 'neighbor-dependent' power mask especially in the non-cooperative *CogMesh* networking.

To implement non-cooperative power allocation in *CogMesh*, one of the most important concern is the definition of the received reward. As mentioned above, a higher SINR at the receiver will generally result in a lower bit error rate and hence a higher throughput. However, achieving a high SINR requires the SU to transmit at a high power level, which in turn causes more power consumption as well as increases the magnitude of the interference for other users, especially the PUs. Accordingly, we choose the average amount of bits received correctly per unit of energy consumption as the reward function to quantify the tradeoff (as in [15]), as this brings practical and meaningful metric to define the energy efficiency,

$$\mathcal{R}_i(p_i, \mathbf{p}_{-i}) = \frac{W \log_2 (1 + \gamma_i(p_i, \mathbf{p}_{-i})/\Gamma)}{p_i}.$$

Here, Γ is the gap between un-coded M-QAM and the capacity, minus the coding gain. And we assume that CR transmitters use variable-rate M-QAM, with a bounded probability of symbol error and trellis coding with a nominal coding gain.

Considering the power mask constraint (4), meanwhile the maximum transmission power level p_i^{\max} , the action set of SU i is then $\mathcal{P}_i = [p_i^{\min}, \bar{p}_i^{\max}]$, where $\bar{p}_i^{\max} = \min(p_i^{\max}, p_{mask})$. We formulate the SUs' selfish behaviors with the theory of non-cooperative game defined by a tuple $\mathcal{G} = \langle \mathcal{N}, \mathcal{P}, \{\mathcal{R}_i(\cdot)\} \rangle$, where $\mathcal{P} = \mathcal{P}_1 \times \dots \times \mathcal{P}_N$ is the action space available for all SUs. Formally, the non-cooperative power allocation game in *CogMesh* can be defined by

$$\begin{aligned} & \max_{p_i \in \mathcal{P}_i} \mathcal{R}_i(p_i, \mathbf{p}_{-i}) \\ & \text{s.t. } \gamma_i \geq \gamma_i^*, \end{aligned}$$

for all $i \in \mathcal{N}$. The solution of this game can be derived in the sense of Nash Equilibrium (NE) [5].

Definition 1: A transmission power vector (p_1^*, \dots, p_N^*) is an NE if, for each SU i ,

$$\mathcal{R}_i(p_i^*, \mathbf{p}_{-i}^*) \geq \mathcal{R}_i(p_i, \mathbf{p}_{-i}^*), \text{ for all } p_i \in \mathcal{P}_i.$$

The following proposition shows the sufficient condition for the existence of an NE in the game [18].

Proposition 1: For any given p_{mask} value, there is an NE in game \mathcal{G} if, for $i = 1, \dots, N$:

- 1) The action set \mathcal{P}_i is a closed and bounded convex set.
- 2) The reward function $\mathcal{R}_i(p_i, \mathbf{p}_{-i})$ is continuous in (p_i, \mathbf{p}_{-i}) and quasi-concave in p_i .

B. Stochastic Power Allocation by Multi-agent Q-learning

The wireless communication system can be considered as a discrete-time system. In this section, we model the SUs' selfish behaviors within stochastic game framework, in which every SU plays the role as an intelligent agent. To be compatible with the multi-agent Q-learning framework, we first discrete the continuous action profile $\mathcal{P}_i = [p_i^{\min}, \bar{p}_i^{\max}]$ as the following

$$p_i(a_i) = \left(1 - \frac{a_i}{m_i}\right) p_i^{\min} + \frac{a_i}{m_i} \bar{p}_i^{\max}, a_i = 0, \dots, m_i.$$

We designate $a_i \in \mathcal{A}_i = \{0, \dots, m_i\}$ as the SU i 's action. Then, it's necessary to identify the environment state, the associated reward and the next state.

1) *State:* Since there is no cooperation among the SUs, the state should be defined based on the local observation of the environment. At time slot t , we can express the state s_i^t observed by the SU i as

$$s_i^t = (i, \mathcal{I}_i, p_i(a_i))_t.$$

Herein, $\mathcal{I}_i \in \{0, 1\}$ specifies whether the SU i 's SINR γ_i at the corresponding receiver end is above or below its threshold γ_i^* . That is,

$$\mathcal{I}_i = \begin{cases} 1, & \text{if } \gamma_i \geq \gamma_i^*; \\ 0, & \text{otherwise.} \end{cases}$$

2) *Reward:* The reward $\mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) = \mathcal{R}_i(a_i, \mathbf{a}_{-i})$ of SU i in state s_i is the immediate return due to the execution of action a_i when all the other SUs choose actions $\mathbf{a}_{-i} = (a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_N)$. Specifically, it is a return of choosing power level $p_i(a_i)$ in state s_i to ensure the transmission QoS requirement as well as to achieve the power efficiency.

3) *Next State*: According to the definition of state s_i^t defined in 1), we can see that the state transition from s_i^t to s_i^{t+1} is determined by the stochastic power allocations of all SUs.

Thus the non-cooperative game \mathcal{G} is converted to the discrete form $\mathcal{G}' = \langle \mathcal{N}, \{A_i\}, \{\mathcal{R}_i\} \rangle$, i.e., each SU chooses the strategy $\pi_i(s_i)$ independently to maximize its total expected discounted reward

$$\max_{\pi_i \in \Pi_i} \left\{ E \left[\sum_{t=0}^{\infty} \beta^t \mathcal{R}_i(s_i^t, \pi_i(s_i^t), \boldsymbol{\pi}_{-i}(s_i^t)) \mid s_i^0 = s_i \right] \right\}, \forall i \in \mathcal{N},$$

where $\boldsymbol{\pi}_{-i}(s_i^t) = (\pi_1(s_1^t), \dots, \pi_{i-1}(s_{i-1}^t), \pi_{i+1}(s_{i+1}^t), \dots, \pi_N(s_N^t))$ and Π_i is the set of strategies available to SU i . A strategy π_i of SU i in state s_i is defined to be a probability vector $\pi_i(s_i) = [\pi_i(s_i, 0), \dots, \pi_i(s_i, m_i)]$, where $\pi_i(s_i, a_i)$ means the probability with which the SU i chooses action a_i when in state s_i . For the case of completely exact information about the other SUs' strategies $\boldsymbol{\pi}_{-i} = (\pi_1, \dots, \pi_{i-1}, \pi_{i+1}, \dots, \pi_N)$, we define the total expected discounted reward of SU i over an infinite time slots as

$$\begin{aligned} & V_i(s_i, \pi_i, \boldsymbol{\pi}_{-i}) \\ &= E \left[\sum_{t=0}^{\infty} \beta^t \mathcal{R}_i(s_i^t, \pi_i(s_i^t), \boldsymbol{\pi}_{-i}(s_i^t)) \mid s_i^0 = s_i \right] \\ &= E [\mathcal{R}_i(s_i, \pi_i(s_i), \boldsymbol{\pi}_{-i}(s_i))] + \beta \sum_{s'_i} T_{s_i s'_i}(\pi_i(s_i), \boldsymbol{\pi}_{-i}(s_i)) V_i(s'_i, \pi_i, \boldsymbol{\pi}_{-i}), \end{aligned}$$

where $T_{s_i s'_i}(\cdot)$ is the state transition probability, and

$$E [\mathcal{R}_i(s_i, \pi_i(s_i), \boldsymbol{\pi}_{-i}(s_i))] = \sum_{a_1 \in \mathcal{A}_1} \cdots \sum_{a_N \in \mathcal{A}_N} \left[\mathcal{R}_i(a_i, \mathbf{a}_{-i}) \prod_{j=1}^N \pi_j(s_j, a_j) \right].$$

In the stochastic power allocation game, each SU behaves as an learning agent whose task is to learn the optimal strategy $\pi_i^*(s_i) (i = 1, \dots, N)$ for each state s_i . Let $\boldsymbol{\pi}_{-i}^* = (\pi_1^*, \dots, \pi_{i-1}^*, \pi_{i+1}^*, \dots, \pi_N^*)$.

Definition 2: A tuple of N strategies $(\pi_i^*, \boldsymbol{\pi}_{-i}^*)$ is an NE if, for each SU i ,

$$V_i(s_i, \pi_i^*, \boldsymbol{\pi}_{-i}^*) \geq V_i(s_i, \pi_i, \boldsymbol{\pi}_{-i}^*), \text{ for all } \pi_i \in \Pi_i.$$

Every finite strategic-form game has a mixed strategy equilibrium [5], that is, there always exists an NE in our game formulation of stochastic power allocation. The optimal strategy

satisfies the Bellman optimality equation, that is, for secondary user i

$$V_i(s_i, \pi_i^*, \boldsymbol{\pi}_{-i}^*) = \max_{a_i \in \mathcal{A}_i} \left\{ E \left[\mathcal{R}_i(s_i, a_i, \boldsymbol{\pi}_{-i}^*(s_i)) \right] + \beta \sum_{s'_i} T_{s_i s'_i}(a_i, \boldsymbol{\pi}_{-i}^*(s_i)) V_i(s'_i, \pi_i^*, \boldsymbol{\pi}_{-i}^*) \right\}, \quad (5)$$

where

$$E \left[\mathcal{R}_i(s_i, a_i, \boldsymbol{\pi}_{-i}^*(s_i)) \right] = \sum_{a_1 \in \mathcal{A}_1} \cdots \sum_{a_{i-1} \in \mathcal{A}_{i-1}} \sum_{a_{i+1} \in \mathcal{A}_{i+1}} \cdots \sum_{a_N \in \mathcal{A}_N} \left[\mathcal{R}_i(a_i, \mathbf{a}_{-i}) \prod_{j=1, j \neq i}^N \pi_j^*(s_j, a_j) \right].$$

We define the optimal Q -value Q_i^* of SU i as the current expected reward plus its future rewards when all SUs follow the Nash equilibrium strategies, that is,

$$Q_i^*(s_i, a_i) = E \left[\mathcal{R}_i(s_i, a_i, \boldsymbol{\pi}_{-i}^*(s_i)) \right] + \beta \sum_{s'_i} T_{s_i s'_i}(a_i, \boldsymbol{\pi}_{-i}^*(s_i)) V_i(s'_i, \pi_i^*, \boldsymbol{\pi}_{-i}^*). \quad (6)$$

Combining equations (5) and (6), it's easy to get

$$Q_i^*(s_i, a_i) = E \left[\mathcal{R}_i(s_i, a_i, \boldsymbol{\pi}_{-i}^*(s_i)) \right] + \beta \sum_{s'_i} T_{s_i s'_i}(a_i, \boldsymbol{\pi}_{-i}^*(s_i)) \max_{b_i \in \mathcal{A}_i} Q_i^*(s'_i, b_i).$$

The multi-agent Q -learning process tries to find $Q_i^*(s_i, a_i)$ in a recursive way using the information $\langle a_i, s_i, s_i, \pi_i^t \rangle$ ($i = 1, \dots, N$), where $s_i (= s_i^t)$ and $s'_i (= s_i^{t+1})$ are the states at time slot t and $t + 1$, respectively; and a_i and π_i^t are the SU i 's action taken at the end of time slot t and the transmission strategy during time slot t . The proposed multi-agent Q -learning rule is

$$Q_i^{t+1}(s_i, a_i) = (1 - \alpha^t) Q_i^t(s_i, a_i) + \alpha^t \left\{ \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) \prod_{j=1, j \neq i}^N \pi_j^t(s_j, a_j) + \beta \max_{b_i \in \mathcal{A}_i} Q_i^t(s'_i, b_i) \right\}. \quad (7)$$

where $\alpha^t \in [0, 1)$ is the learning rate.

An intuitive explanation for Equation (7) is that, once the power level $p_i(a_i)$ is selected, the increasing quantity in the corresponding Q -value is updated by combining the old value and the new expected reward. More specifically, given the probabilities $\{\pi_j^t(s_j, a_j)\}_{j=1, j \neq i}^N$ of the other SUs choosing power levels $\{p_j(a_j)\}_{j=1, j \neq i}^N$, if the SU i achieves higher reward $\mathcal{R}_i(a_i, \mathbf{a}_{-i})$ when selecting power level $p_i(a_i)$, then the $Q_i^t(s_i, a_i)$ -value is increased by a higher value. Notice that the proposed multi-agent Q -learning algorithm not only needs the SU i 's own information, but the strategies of the other SUs. However, in this paper, the strategy is myopic since we assume that there is no cooperation among the SUs.

C. Design Objective and Challenging Issues

Our aim is to design stochastic power allocation in non-cooperative *CogMesh* with self-interested SUs. The reward of each SU is a function of the joint actions of all SUs. Accordingly, we apply the multi-agent Q -learning approach to model the interaction among the SUs' strategy decisions. Rather than choosing the best transmission power level, a SU in the stochastic learning process chooses the best mixed strategy. The problem is challenging due to the fact that every SU may not be aware of the following two things during the learning process:

- 1) the number of SUs coexist in the system;
- 2) strategies available to the other SUs.

The SU can only observe its own information, such as the environment state, the strategy, and the received rewards.

From Equation (7), in order to learn the optimal strategy, SU i needs to know not only its own strategy, but also the other SUs' transmission strategies $\pi_j^t (j \in \mathcal{N} \setminus i)$. Along with the discussion, we see that the obtained multi-agent Q -learning algorithm cannot solve the power allocation problem directly because no SU can observe the competing SUs' private information in a non-cooperative *CogMesh* networking scenario. Therefore, the challenging problem arises: ***how to design a stochastic non-cooperative power allocation scheme that guarantees SUs learning the optimal strategies with only private and incomplete information?***

IV. STOCHASTIC POWER ALLOCATION WITH CONJECTURE BASED MULTI-AGENT Q -LEARNING APPROACH

As discussed in the previous section, the main disadvantage of the derived multi-agent Q -learning algorithm is its requirement to account for the competing SUs' strategy information. In non-cooperative power control, however, the SUs only know what reward they are getting from their current strategy. In this section, we propose a stochastic non-cooperative power allocation scheme with private and incomplete information. To make the multi-agent Q -learning algorithm sensible in non-cooperative *CogMesh* networking environment, it is clear that the SU needs to conjecture the other SUs' strategy decisions without any coordination among the local clusters [24]. This motivates the conjecture based multi-agent Q -learning.

A. Individual Behavior and Evolution

The goal of this paper is to design a simple non-cooperative power allocation algorithm that requires quite limited information exchanges among the SUs. In game-theoretic point of view, the reached NE is based on the assumptions about what knowledge the SUs possess and assumes that every SU's strategy will not change at the NE. Therefore, the SUs operating at the NE can be viewed as learning agents behaving optimally with respect to their conjectures about the strategies of the other SUs.

From Equation (7), we can see that the SU i 's current expected reward depends on both its own decision and the other SUs' transmission policies. However, in the non-cooperative scenario, it is hard for the SUs to obtain the information of exact transmission strategies of their competitors. We define $c_i^t(s_i, a_i) = \prod_{j=1, j \neq i}^N \pi_j^t(s_j, a_j)$ for the SU i in time slot t , to be the conjecture representing the aggregated effect on the $Q_i^{t+1}(s_i, a_i)$ -value when all the other SUs choosing actions \mathbf{a}_{-i} according to their corresponding strategies $\boldsymbol{\pi}_{-i}^t(s_i) = (\pi_1^t(s_1), \dots, \pi_{i-1}^t(s_{i-1}), \pi_{i+1}^t(s_{i+1}), \dots, \pi_N^t(s_N))$. Therefore, we assume that $c_i^t(s_i, a_i)$ is the only information that the SU i has about the contention level of the entire *CogMesh* networking environment, because it is a metric that the SU i can easily calculate based on local observations.

Specifically, from SU i 's viewpoint, the probability of experiencing environment state s_i' is $\zeta_i = \pi_i^t(s_i, a_i)c_i^t(s_i, a_i)$. In other words, the probability that the SU i receives reward $\mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i})$ is ζ_i . Let n_i denote the number of time slots between any two consecutive slot that SU i achieves the same reward $\mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i})$, then n_i has an independent and identical distribution (i.i.d.) with ζ_i . Thereupon, we have $\zeta_i \cong 1/(1+\bar{n}_i)$, where \bar{n}_i is the mean value of n_i and can be locally computed by the SU i itself through the observation of its reward history. Since SU i knows its own transmission strategy $\pi_i^t(s_i, a_i)$, it can estimate $c_i^t(s_i, a_i)$ through $\tilde{c}_i^t(s_i, a_i) = 1/[(1+\bar{n}_i)\pi_i^t(s_i, a_i)]$. Note that the action available to SU i is to choose the transmission power level according to strategy $\pi_i^t(s_i)$. We can express the SU i 's conjecture $\tilde{c}_i^t(s_i, a_i)$ as a function of its own transmission strategy. A simple method is to deploy the linear model, i.e.,

$$\tilde{c}_i^t(s_i, a_i) = \bar{c}_i(s_i, a_i) - \omega_i^{s_i, a_i} [\pi_i^t(s_i, a_i) - \bar{\pi}_i(s_i, a_i)], \quad (8)$$

where the so-called reference points [13], $\bar{c}_i(s_i, a_i)$ and $\bar{\pi}_i(s_i, a_i)$, are specific conjecture and probability, and $\omega_i^{s_i, a_i}$ is a positive scalar. In this paper, the reference points are considered as exogenously given and of common knowledge. That is, SU i assumes that the other SUs will

observe its deviation from its reference point $\bar{\pi}_i(s_i^t, a_i)$ and the aggregate effect deviates from the reference point $\bar{c}_i(s_i, a_i)$ by a quantity proportional to the deviation of $\pi_i^t(s_i, a_i) - \bar{\pi}_i(s_i, a_i)$.

Among different choices for capturing the impact of the competing SUs as a function of its own strategy, the linear model shown in Equation (8) is the simplest form one can think of. In the following, we will show that such simple model is sufficient for the secondary users to achieve optimal transmissions. The critical question is how to choose the parameters $\{\bar{c}_i(s_i, a_i), \bar{\pi}_i(s_i, a_i), \omega_i^{s_i, a_i}\}$ to achieve the optimal strategies π_i^* . We can consider setting the parameter in Equation (8) to be:

$$\omega_i^{s_i, a_i} = \frac{\prod_{j=1, j \neq i}^N \pi_j^*(s_j, a_j)}{\pi_i^*(s_i, a_i)}.$$

It's very easy to verify that, if the reference points are $\bar{c}_i(s_i, a_i) = \prod_{j=1, j \neq i}^N \pi_j^*(s_j, a_j)$ and $\bar{\pi}_i(s_i, a_i) = \pi_i^*(s_i, a_i)$, we have $\tilde{c}_i^*(s_i, a_i) = \prod_{j=1, j \neq i}^N \pi_j^*(s_j, a_j)$. Therefore, such configuration of the conjectures \tilde{c}_i^* and the strategies π_i^* achieve the optimal transmission. In non-cooperative learning scenarios, SUs learn when they modify their conjectures based on the new observations. Specifically, we first allow the SUs to revise their reference points based on their past local observations. We propose a simple rule for the SUs to update their reference points. In time slot t , the SU i set $\bar{c}_i(s_i, a_i)$ and $\bar{\pi}_i(s_i, a_i)$ to be $c_i^{t-1}(s_i, a_i)$ and $\pi_i^{t-1}(s_i, a_i)$. That is, Equation (8) becomes

$$\tilde{c}_i^t(s_i, a_i) = c_i^{t-1}(s_i, a_i) - \omega_i^{s_i, a_i} [\pi_i^t(s_i, a_i) - \pi_i^{t-1}(s_i, a_i)], \quad (9)$$

for $i \in \mathcal{N}$.

B. Conjecture based Q-value Updating

Eventually, the multi-agent Q-learning updating rule in Equation (7) is modified as following,

$$Q_i^{t+1}(s_i, a_i) = (1 - \alpha^t)Q_i^t(s_i, a_i) + \alpha^t \left\{ \tilde{c}_i^t(s_i, a_i) \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) + \beta \max_{b_i \in \mathcal{A}_i} Q_i^t(s_i', b_i) \right\}. \quad (10)$$

The SU i updates its Q-values only with its own information using Equation (10) during the stochastic learning process. To avoid observing the other SUs' private strategy information, the SU i conjectures about how its competitors' strategy decisions vary in response to its own actions.

The purpose of stochastic power allocation is to improve performance by explicitly balancing two competing objectives: 1) searching for better transmission power level (exploration) and 2) gathering as much reward as possible (exploitation), such that the SU not only reinforces the evaluation of the power level it already knows to be good but also explores new one. Though ϵ -greedy selection [7] is an efficient method of balancing exploration and exploitation in reinforcement learning. One drawback is that it chooses equally among all available actions when it explores. This implies that the worst action is as likely to be chosen as the best one.

An alternative solution is to vary the action probabilities as a graded function of the Q -value. The greedy action is given the highest selection probability, but all the others are ranked and weighted according to their Q -values. The most common method is to use a Boltzmann distribution. The SU i chooses action a_i in state s_i at time step t with probability [20],

$$\pi_i^t(s_i, a_i) = \frac{e^{Q_i^t(s_i, a_i)/\tau}}{\sum_{b \in \mathcal{A}_i} e^{Q_i^t(s_i, b)/\tau}}, \quad (11)$$

where τ is a positive parameter called the temperature. High temperatures cause the action probabilities to be all nearly equal. Low temperatures cause big difference in selection probabilities for actions differ in their Q -values.

Now, the steps concerning power allocation corresponding to the conjecture-based multi-agent Q -learning algorithm are summarized as follows:

Algorithm: Conjecture based Multi-agent Q -learning Algorithm for SU i

Initialization:

Let $t = 0$,

For each s_i, a_i **Do**

Initialize strategy $\pi_i^t(s_i, a_i)$, Q -values $Q_i^t(s_i, a_i)$, and the parameter $\omega_i^{s_i, a_i} > 0$.

End For

Evaluate the initial state $s_i = s_i^t$.

Learning:

Loop

- (1) Choose action a_i according to $\pi_i^t(s_i)$.
- (2) Measure the SINR γ_i with the feedback information of the intended secondary receiver. Construct the current environment state $s'_i = s_i^{t+1}$ by identifying the transmission power level, and comparing γ_i with the threshold γ_i^* .
- (3) If $\gamma_i \geq \gamma_i^*$, then a reward $\mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i})$ can be achieved; otherwise, the receiver can not receive correctly, thus obtains zero reward.
- (4) Update $Q_i^{t+1}(s_i, a_i)$ based on $\tilde{c}_i^t(s_i, a_i)$ according to $Q_i^{t+1}(s_i, a_i) = (1 - \alpha^t) Q_i^t(s_i, a_i) + \alpha^t \left\{ \tilde{c}_i^t(s_i, a_i) \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) + \beta \max_{b_i \in \mathcal{A}_i} Q_i^t(s'_i, b_i) \right\}$.
- (5) Update the strategy $\pi_i^{t+1}(s_i, a_i) = e^{Q_i^{t+1}(s_i, a_i)/\tau} / \sum_{b_i \in \mathcal{A}_i} e^{Q_i^{t+1}(s_i, b_i)/\tau}$, for all $a_i \in \mathcal{A}_i$.
- (6) Update the conjecture $\tilde{c}_i^{t+1}(s_i, a_i) = c_i^t(s_i, a_i) - \omega_i^{s_i, a_i} [\pi_i^{t+1}(s_i, a_i) - \pi_i^t(s_i, a_i)]$.
- (7) $s_i = s_i^{t+1}$.

End Loop

Next, we are interested in the convergence of this algorithm. Our proof relies on the following lemma by Szepesvari and Littman [21], which establishes the convergence of a general Q -learning process updated by a pseudo-contraction operator. Let \mathcal{Q} be the space of all Q -values.

Lemma: Assume that α^t in Equation (10) satisfies the sufficient conditions of Theorem in [23], and the mapping $\mathcal{H}^t : \mathcal{Q} \rightarrow \mathcal{Q}$ meets the following condition: there exists a number $0 < \lambda < 1$ and a sequence $\xi^t \geq 0$ converging to zero w.p. 1, such that $\| \mathcal{H}^t Q^t - \mathcal{H}^t Q^* \| \leq \lambda \| Q^t - Q^* \| + \xi^t$ for all $Q^t \in \mathcal{Q}$ and $Q^* = E[\mathcal{H}^t Q^*]$, then the iteration defined by

$$Q^{t+1} = (1 - \alpha^t) Q^t + \alpha^t (\mathcal{H}^t Q^t),$$

converges to Q^* w.p. 1.

For an N -player stochastic game, we define the operator \mathcal{H}^t as follows.

Definition 3: Let $Q^t = (Q_1^t, \dots, Q_N^t)$, where $Q_i^t \in \mathcal{Q}_i$ for $i = 1, \dots, N$, and $\mathcal{Q} = \mathcal{Q}_1 \times \dots \times \mathcal{Q}_N$. $\mathcal{H}^t : \mathcal{Q} \rightarrow \mathcal{Q}$ is a mapping on the complete metric space \mathcal{Q} into \mathcal{Q} , $\mathcal{H}^t Q^t = (\mathcal{H}^t Q_1^t, \dots, \mathcal{H}^t Q_N^t)$, where

$$\mathcal{H}^t Q_i^t = \tilde{c}_i^t(s_i, a_i) \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) + \beta \max_{b_i \in \mathcal{A}_i} Q_i^t(s'_i, b_i).$$

Then we proceed to prove that $Q^* = E[\mathcal{H}^t Q^*]$.

Proposition 2: For an N -player stochastic game, $Q^* = E[\mathcal{H}^t Q^*]$, where $Q^* = (Q_1^*, \dots, Q_N^*)$.

Proof: Since

$$\begin{aligned} Q_i^*(s_i, a_i) &= E[\mathcal{R}_i(s_i, a_i, \pi_{-i}^*(s_i))] + \beta \sum_{s'_i} T_{s_i s'_i}(a_i, \pi_{-i}^*(s_i)) \max_{b_i \in \mathcal{A}_i} Q_i^*(s'_i, b_i) \\ &= \sum_{s'_i} T_{s_i s'_i}(a_i, \pi_{-i}^*(s_i)) \left\{ \prod_{j=1, j \neq i}^N \pi_j^*(s_j, a_j) \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) + \beta \max_{b_i \in \mathcal{A}_i} Q_i^*(s'_i, b_i) \right\}. \end{aligned}$$

From Equation (9), $\tilde{c}_i^*(s_i, a_i) = \prod_{j=1, j \neq i}^N \pi_j^*(s_j, a_j)$. Thus,

$$Q_i^*(s_i, a_i) = E[\mathcal{H}^t Q^*(s_i, a_i)],$$

for all s_i and a_i . ■

We further define the distance between two Q -values.

Definition 4: For any $Q, Q' \in \mathcal{Q}$, we define

$$\|Q - Q'\| \doteq \max_i \max_{s_i} \max_{a_i} |Q_i(s_i, a_i) - Q'_i(s_i, a_i)|.$$

Proposition 3: \mathcal{H}^t is a contraction mapping operator.

Proof: The proof is given in Appendix.

We can now present our main result in this paper that the learning process induced by **Algorithm** converges.

Theorem: Regardless of any initial value chosen for $Q_i^0(s_i, a_i)$, if τ is sufficiently large, **Algorithm** converges.

Proof: The proof is the direct application of *Lemma*, which establishes the convergence given two conditions. First, \mathcal{H}^t is a contraction mapping operator, by *Proposition 3*. Second, the fixed point condition, $Q^* = E[\mathcal{H}^t Q^*]$, is ensured by *Proposition 2*. Therefore, the learning process expressed by Equation (10) converges.

V. NUMERICAL RESULTS

To demonstrate the performance of the proposed conjecture based multi-agent Q -learning algorithm, we present simulation experiments of a hybrid *CogMesh* consisting of one PU network and one CR network. Users in *CogMesh* are uniformly distributed over a $300\text{m} \times 300\text{m}$ square

area, and share the same frequency band with bandwidth of $W = 1\text{MHz}$. The links can communicate directly if the distance between transmitter and the corresponding receiver is no more than 30m. The time is divided into slots, each of length 10ms. During each time slot, each PU attempts to transmit with a probability of κ , the PU's behavior factor. It's supposed that the PUs have only one transmission power level of 200mW, the AWGN power $\sigma = 10^{-7}\text{mW}$, and $\Gamma = 1$. Also, we set the power mask to be 200mW for all SUs. The link gains used in this paper are given by

$$h = KF \left(\frac{d}{d_0} \right)^{-n}, \text{ for } d > d_0,$$

where K is a constant set to be 10^{-6} , the shadowing factor F is a random number and is independent and identically generated from a lognormal distribution with a mean of 0dB and variance 6dB, d is the physical distance between transmitter and receiver, d_0 is the reference distance, and n is the path loss exponent. In the whole simulation process, we set $d_0 = 1$ and $n = 4$. And we here point out that all simulated curves in this paper show the average over 200 episodes.

As for the proposed conjecture based multi-agent Q -learning algorithm, it's implemented by each SU with a discount factor $\beta = 0.9$. And we use the following learning rate

$$\alpha^t = \frac{\alpha^0}{\theta^t},$$

where $\alpha^0 \in [0, 1)$ is the initial learning rate, and $\theta > 1$ is a scalar. Like any other learning scheme, the SUs need a learning phase to learn the optimal transmission strategies under the assumption that each SU can perfectly conjecture the probability $\prod_{j=1, j \neq i}^N \pi_j^t(s_j, a_j)$ during each time slot. However, once the strategies are acquired, the SUs take only one iteration to reach the optimal energy-efficient transmission configuration, when starting at any initial environment states $s_i (i = 1, \dots, N)$. The major concern for our proposed algorithm is the convergence speed of the stochastic learning dynamics. We first simulate a relatively simple networking scenario consisting of three pairs of SU links coexisting with three pairs of PU links with a behavior factor $\kappa = 0.5$. The SUs have two transmission power levels $\{100\text{mW}, 200\text{mW}\}$. That is, in the proposed algorithm, $m_i = 1$ and $\mathcal{N} = \{1, 2\}$.

Without the loss of generality, we take SU 1 for example. Fig. 3 and Fig. 4 show the simulation results for different α^0 and τ , which indicate that the proposed algorithm converges. We can also

see from the Fig. 3 that larger τ results in worse expected reward. This is because exploration lasts for a longer time even if the best power level achieving optimal transmission was already visited. Thus, during the learning process, the SU should set a sufficiently large temperature to balance the tradeoff between exploration and exploitation or has to dynamically adjust it. The curves in Fig. 4 illustrate that when τ is small, for smaller α^0 the convergence performance is worse. Since the Q -values converges slowly, then still exploration phases dominates the learning procedure, which may lead to decreasing the opportunities of achieving optimal transmission configuration on average. Overall, the performance of our proposed algorithm is good when choosing a suitable learning rate α^0 . If the algorithm is deployed by the SUs in *CogMesh* environment, α^0 has to be chosen in advance.

Next, for a more general case, we consider that the CR network consists of six SUs co-locating with five PUs. The PUs attempt to transmit with a probability $\kappa = 1$. Each SU has multiple transmission power levels. The discrete transmission power levels the SUs used are in the range from 100mW to 200mW equally spaced by 20mW. We compare the expected rewards of SUs achieved by the proposed algorithm with the system's optimum $\mathcal{R}_i^{opt} = \max_{\mathbf{p}} \mathcal{R}_i(\mathbf{p})$ in Fig. 5. It can be seen from the graph that the achieved performance is close to the optimum and the performance loss is no more than 25% on the average.

Fig. 6 depicts the expected rewards of the six secondary users versus the PU's behavior factor κ under the same networking environment assumptions as in Fig. 5. As expected, a higher κ results in higher interference caused by the PUs to the SUs, i.e., the expected rewards are degraded.

VI. CONCLUSION

In this paper, we have studied the non-cooperative power allocation problem specifically in *CogMesh* which is modeled as a stochastic learning process. We extend the single-agent Q -learning algorithm to a multi-user context. Due to the non-cooperation among the local clusters, a conjecture based multi-agent Q -learning approach is proposed to reach the optimal transmission strategies with only private and incomplete information. The learning SU performs Q -function updating based on the conjecture about other SUs' behaviors over the current Q -values. This learning algorithm provably converges given certain restrictions that arise during learning procedure, and the simulations demonstrate the effectiveness of the algorithm to improve

energy efficiency. The results in this paper provide us with a new approach to design the protocols for the non-cooperative CR networks.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions, which have helped improve the quality of this paper.

APPENDIX

Proof of Proposition 3.

Proof:

$$\begin{aligned}
& \|\mathcal{H}^t Q - \mathcal{H}^t Q'\| \\
&= \max_i \max_{s_i} \max_{a_i} |\mathcal{H}^t Q_i(s_i, a_i) - \mathcal{H}^t Q'_i(s_i, a_i)| \\
&= \max_i \max_{s_i} \max_{a_i} \left| [\tilde{c}_i(s_i, a_i) - \tilde{c}'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) + \right. \\
&\quad \left. \beta \left[\max_{b_i \in \mathcal{A}_i} Q_i(s'_i, b_i) - \max_{b_i \in \mathcal{A}_i} Q'_i(s'_i, b_i) \right] \right| \\
&\leq \max_i \max_{s_i} \max_{a_i} \left| [\tilde{c}_i(s_i, a_i) - \tilde{c}'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) \right| + \\
&\quad \max_i \max_{s_i} \beta \left| \max_{b_i \in \mathcal{A}_i} Q_i(s'_i, b_i) - \max_{b_i \in \mathcal{A}_i} Q'_i(s'_i, b_i) \right| \\
&\leq \max_i \max_{s_i} \max_{a_i} |[\tilde{c}_i(s_i, a_i) - \tilde{c}'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i})| + \beta \|Q - Q'\|.
\end{aligned}$$

We discuss the first item $[\tilde{c}_i(s_i, a_i) - \tilde{c}'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i})$ in the last inequality above. Due to the fact that the reference points are exogenously given and of common knowledge, then we have

$$[\tilde{c}_i(s_i, a_i) - \tilde{c}'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) = -\omega_i^{s_i, a_i} [\pi_i(s_i, a_i) - \pi'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) \quad (\text{A-1})$$

We first concentrate on the item $\pi_i(s_i, a_i)$ in Equation (A-1). By applying Equation (11), we have

$$\pi_i(s_i, a_i) = \frac{e^{Q_i(s_i, a_i)/\tau}}{\sum_{b \in \mathcal{A}_i} e^{Q_i(s_i, b)/\tau}}.$$

When τ is sufficiently large, we get

$$e^{Q_i(s_i, a_i)/\tau} = 1 + \frac{Q_i(s_i, a_i)}{\tau} + \vartheta \left(\frac{Q_i(s_i, a_i)}{\tau} \right),$$

where $\vartheta\left(\frac{Q_i(s_i, a_i)}{\tau}\right)$ is a polynomial of order $\mathcal{O}\left(\left(\frac{Q_i(s_i, a_i)}{\tau}\right)^2\right)$. It's very easy to verify that

$$\pi_i(s_i, a_i) = \frac{1}{m_i + 1} + \frac{Q_i(s_i, a_i)}{(m_i + 1)\tau} + \varrho(\{Q_i(s_i, b)\}_b), \quad (\text{A-2})$$

where $\varrho(\{Q_i(s_i, b)\}_b)$ is the polynomial of smaller order than $\mathcal{O}\left(\frac{Q_i(s_i, a_i)}{\tau}\right)$. Note that the coefficient of the polynomial is independent of the Q -value. Similarly,

$$\pi'_i(s_i, a_i) = \frac{1}{m_i + 1} + \frac{Q'_i(s_i, a_i)}{(m_i + 1)\tau} + \varrho(\{Q'_i(s_i, b)\}_b). \quad (\text{A-3})$$

Substituting Equations (A-2) and (A-3) to Equation (A-1) establishes

$$\begin{aligned} & [\tilde{c}_i(s_i, a_i) - \tilde{c}'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) \\ &= -\omega_i^{s_i, a_i} \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i}) \left[\frac{Q_i(s_i, a_i)}{(m_i + 1)\tau} - \frac{Q'_i(s_i, a_i)}{(m_i + 1)\tau} + \varrho(\{Q_i(s_i, b)\}_b) - \varrho(\{Q'_i(s_i, b)\}_b) \right] \\ &= -C_i(s_i, a_i) \left\{ \frac{1}{m_i + 1} \left[\frac{Q_i(s_i, a_i)}{\tau} - \frac{Q'_i(s_i, a_i)}{\tau} \right] + \varrho(\{Q_i(s_i, b)\}_b) - \varrho(\{Q'_i(s_i, b)\}_b) \right\}. \end{aligned}$$

That is, we can always take a sufficiently large τ such that

$$|[\tilde{c}_i(s_i, a_i) - \tilde{c}'_i(s_i, a_i)] \mathcal{R}_i(s_i, a_i, \mathbf{a}_{-i})| \leq \frac{1 - m_i\beta}{m_i + 1} \|Q_i(s_i, a_i) - Q'_i(s_i, a_i)\|,$$

which implies

$$\begin{aligned} \|\mathcal{H}^t Q - \mathcal{H}^t Q'\| &\leq \max_i \max_{s_i} \max_{a_i} \frac{\beta}{m_i + 1} \|Q_i(s_i, a_i) - Q'_i(s_i, a_i)\| + \beta \|Q - Q'\| \\ &= \frac{1 + \beta}{m_i + 1} \|Q - Q'\|. \end{aligned}$$

Therefore, \mathcal{H}^t is a contraction mapping operator. This concludes the proof. ■

REFERENCES

- [1] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Networks*, Jan. 2009.
- [2] T. Chen, H. Zhang, G. M. Maggio, and I. Chlamtac, "CogMesh: A cluster-based cognitive radio network," in *Proc. IEEE DySPAN*, Dublin, April 2007, pp. 168–178.
- [3] F. Fu and M. van der Schaar, "Learning to compete for resources in wireless stochastic games," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1904–1919, May 2009.
- [4] —, "Learning to compete for resources in wireless stochastic games," *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 1904–1919, May 2009.
- [5] D. Fudenberg and J. Tirole, *Game Theory*. Cambridge, MA: MIT Press, 1992.

- [6] S. Gao, L. Qian, and D. Vaman, "Distributed energy efficient spectrum access in cognitive radio wireless ad hoc networks," *IEEE Transaction on Wireless Communications*, vol. 8, no. 10, pp. 5202–5213, Oct. 2009.
- [7] E. R. Gomes and R. Kowalczyk, "Dynamic analysis of multiagent Q -learning with ϵ -greedy exploration," in *International Conference on Machine Learning*, 2009.
- [8] A. Greenwald and K. Hall, "Correlated- Q learning," in *ICML 2003*, 2003.
- [9] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE Journal of Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, Feb. 2005.
- [10] Y. T. Hou, Y. Shi, and H. D. Sherali, "Optimal spectrum sharing for multi-hop software defined radio networks," in *INFOCOM 2007*, May 2007, pp. 1–9.
- [11] J. Hu and M. P. Wellman, "Nash Q -learning for general-sum stochastic games," *Journal of Machine Learning Research* 4, pp. 1039–1069, 2003.
- [12] L. L. J. Miettinen and R. Schober, "Distributed transmit power allocation for relay-assisted cognitive-radio systems," *IEEE Transactions on Wireless Communications*, vol. 8, pp. 5187–5201, Oct. 2009.
- [13] A. Jean-Marie and M. Tidball, "Adapting behaviors through a learning process," *Journal of Economic Behavior and Organization*, vol. 60, pp. 399–422, 2006.
- [14] H. Li, "Multiagent Q -learning for aloha-like spectrum access in cognitive radio systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2010, 2010.
- [15] F. Meshkati, M. Chiang, H. V. Poor, and S. C. Schwartz, "A game-theoretic approach to energy-efficient power control in multicarrier cdma systems," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 6, pp. 1115–1129, June 2006.
- [16] J. Mitola and G. Q. Maguire, "Cognitive radios: Making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, Aug. 1999.
- [17] Federal Communications Commission, "Spectrum policy task force," *Rep. ET Docket*, no. 02-135, Nov. 2002.
- [18] C. U. Saraydar, N. B. Mandayam, and D. J. Goodman, "Efficient power control via pricing in wireless data networks," *IEEE Trans. Commun.*, vol. 50, no. 2, pp. 291–303, Feb. 2002.
- [19] Y. Shi and T. Hou, "A distributed optimization algorithm for multi-hop cognitive radio networks," in *Proc. IEEE INFOCOM*, Phoenix, AZ, April 2008, pp. 1292–1300.
- [20] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 1998.
- [21] C. Szepesvari and M. L. Littman, "A unified analysis of value-function-based reinforcement learning algorithms," *Neural Computation*, vol. 11, no. 8, pp. 2017–2060, Nov. 1999.
- [22] F. Wang, M. Krunz, and S. Cui, "Price-based spectrum management in cognitive radio networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, pp. 74–87, Feb. 2008.
- [23] C. J. C. H. Watkins and P. Dayan, " Q -learning," *Machine learning*, vol. 8, pp. 279–292, 1992.
- [24] M. P. Wellman and J. Hu, "Conjectural equilibrium in multiagent learning," *Machine Learning*, vol. 33, pp. 179–200, 1998.
- [25] Y. Wu and D. H. TSANG, "Distributed power allocation algorithm for spectrum sharing cognitive radio networks with QoS guarantee," in *Proc. INFOCOM*, April 2009, pp. 981–989.
- [26] Y. Xing and R. Chandramouli, "Stochastic learning solution for distributed discrete power control game in wireless data networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 4, pp. 932–944, Aug. 2008.

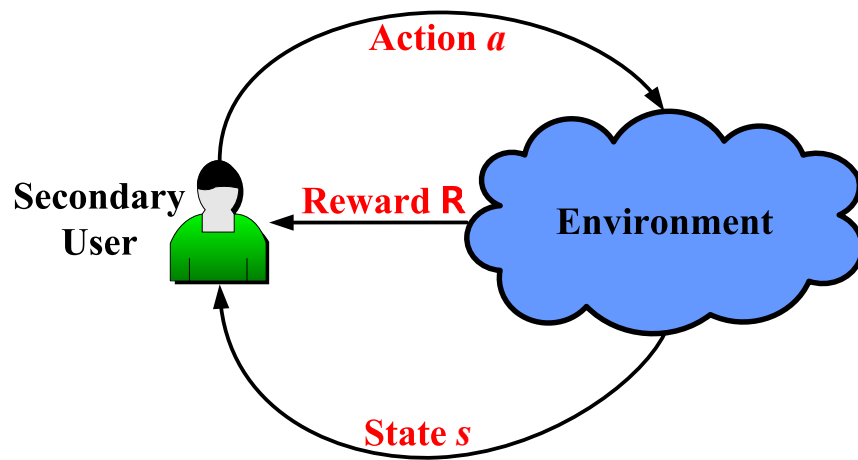


Fig. 1. Reinforcement learning.

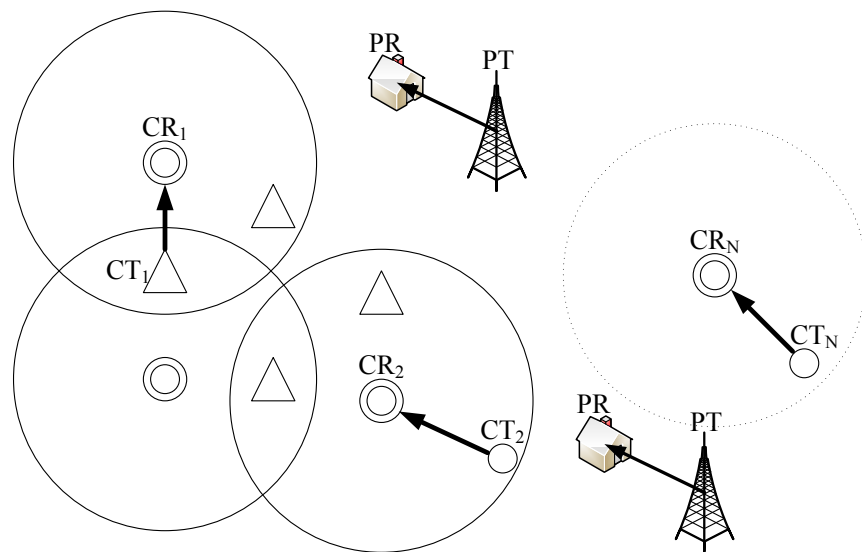


Fig. 2. Cognitive wireless mesh networking (*CogMesh*) scenarios.

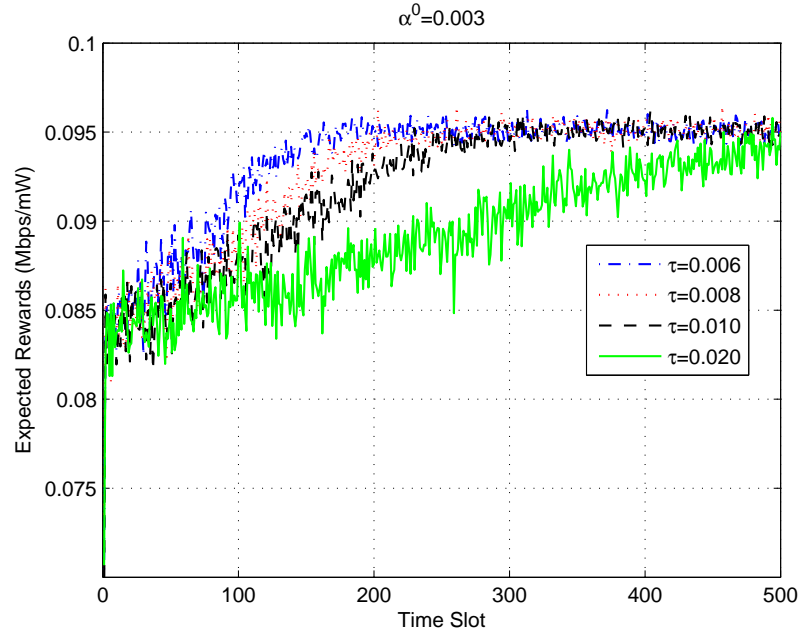


Fig. 3. Performance, when $\kappa = 0.5$: Impact of the temperature τ to expected rewards achieved by SU 1.

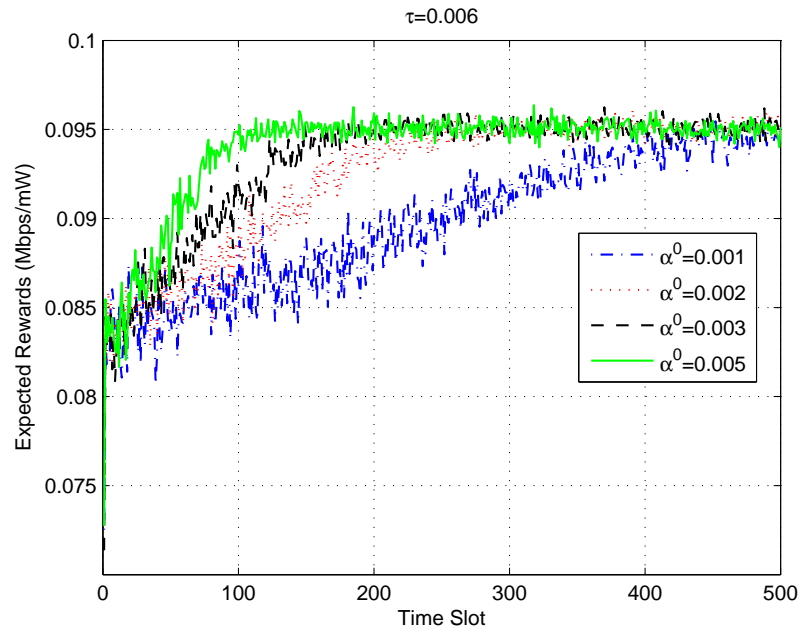


Fig. 4. Performance, when $\kappa = 0.5$: Impact of the learning rate α^0 to expected rewards achieved by SU 1.

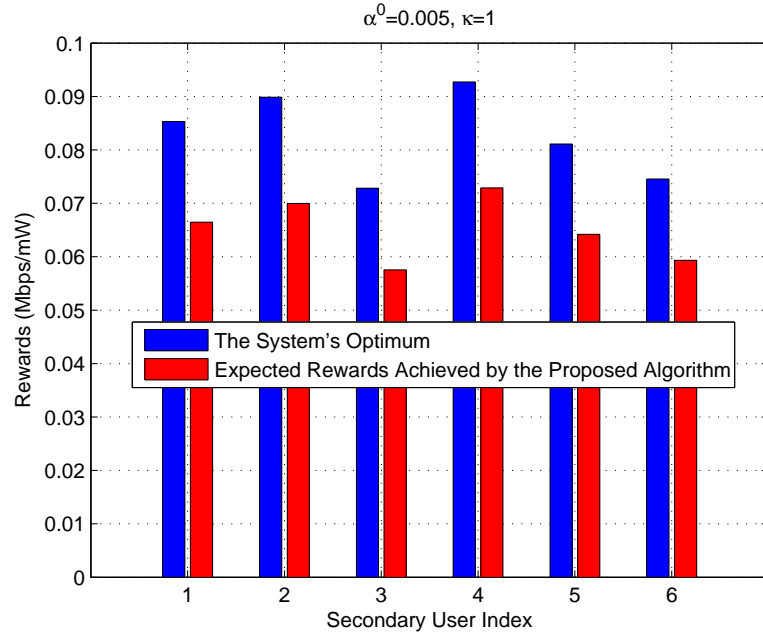


Fig. 5. Performance comparison between the proposed algorithm and the system's optimum.

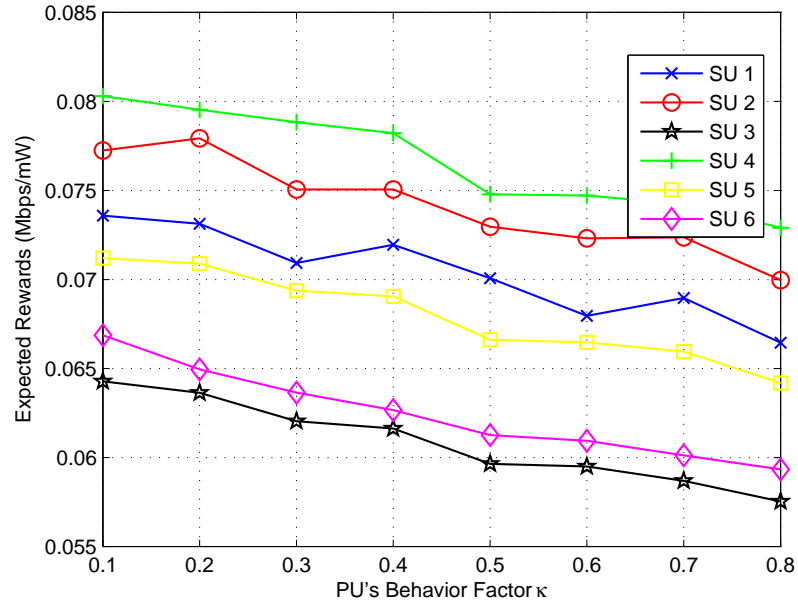


Fig. 6. The expected rewards of the SUs versus the PU's behavior factor κ .